

# MULTI-DYNAMISCHE ARCHITECTUUR

## HOE GAAT DE ARCHITECT OM MET EEN IT-OMGEVING DIE STEEDS MEER DIVERS IS?

**DE WERELD IS MULTIPOLAIR. OOK DE IT-WERELD. STREVEN NAAR EENVORMIGHEID OVER DE GEHELE ORGANISATIE IS EEN SLECHT IDEE. DAT GELDT OOK VOOR IT-ARCHITECTUUR. DIFFERENTIATIE IN ARCHITECTUURKADERS IS NOODZAKELIJK. DIT BETOGEN DE AUTEURS IN MEI IN HET ARTIKEL 'MULTIDYNAMISCHE ARCHITECTUUR: DE WERELD IS NIET EENVORMIG OF BIPOLAIR' DAT OP AG CONNECT TE VINDEN IS. MAAR HOE KRIJGT DIT CONCREET HANDEN EN VOETEN? EEN GOEDE START IS HET DIFFERENTIËREN MET ARCHITECTUURPRINCIPES. IN DIT ARTIKEL GEVEN DE AUTEURS AAN HOE ARCHITECTEN MET BEHULP VAN PRINCIPES EEN EERSTE STAP KUNNEN ZETTEN OM EFFECTIEF MET DE GROTE DIVERSITEIT IN IT OM TE GAAN.**

Door Marlies van Steenberghe, Ton Eusterbrock en Jan Bouman

### DIFFERENTIËREN MET PRINCIPES

Architectuurprincipes zijn een krachtig communicatie- en stuurmiddel. Ze vormen duidelijke uitspraken waaraan wel of niet voldaan wordt. In de praktijk kunnen principes sterk verschillen in concreetheid en precisie. Deze verschillen leiden vaak tot discussie. Bijvoorbeeld wat is een beter principe? Alle data-uitwisseling moet veilig zijn of alle elektronische berichten moeten worden versleuteld met P2PE? Het tweede alternatief is een mogelijke invulling van het eerste alternatief. Wat is het juiste niveau? Als architectuurvoorschriften te specifiek zijn geformuleerd bergt dat een aantal gevaren in zich. Zodra er iets verandert is de kans groot dat de voorschriften niet meer passen. In onvoorziene situaties zijn voorschriften veelal niet voorhanden. En als alles in detail is voorgeschreven

wennen medewerkers eraan de voorschriften klakkeloos toe te passen, ook als dat niet verstandig is. Aan de andere kant, een kleine verzameling abstracte principes geeft in veel situaties te weinig sturing.

Deze potentiële zwakte van principes wordt een kracht als organisaties leren om bewust te sturen met deze niveauverschillen. Interessant in dit verband is het onderscheid tussen een principegebaseerde aanpak en een regelgebaseerde aanpak dat gemaakt wordt in de wereld van regelgeving en toezichthouders. Kort gezegd geven principes een gewenste uitkomst weer (data-uitwisseling moet veilig zijn), terwijl regels aangeven hoe een gewenste uitkomst gerealiseerd moet worden (versleuteling van berichten volgens een voorgeschreven protocol). Met name in Europa is er sprake van een verschuiving van regelgebaseerd naar principegebaseerd toezicht. Als gevolg daarvan neemt de eigen verantwoordelijkheid van bedrijven en instellingen toe en verandert de rol van de toezichthouder.

Dit onderscheid tussen principes en regels is ook toepasbaar op IT-architectuur. Tabel 1 illustreert dat binnen

een verzameling principes het vaak mogelijk is om een keten van principes te herkennen. Soms komen in dezelfde organisatie parallelle ketens voor. Als dit het gevolg is van een gebrek aan coördinatie kan het leiden tot inconsistenties. Maar het kan ook bewust als instrument gebruikt worden.

In de praktijk bevinden architectuurvoorschriften zich op een continuüm met aan het ene uiteinde abstracte principes en aan het andere uiteinde concrete regels. In een regelgerichte architectuur formuleren de centrale architecten alle niveaus van voorschriften. In een principegerichte architectuur beperken de centrale architecten zich tot de top van de hiërarchie en is het formuleren van de lagere principes de verantwoordelijkheid van andere organisatieonderdelen. Daarbij kunnen onderdelen mogelijk verschillende keuzes maken, of kunnen ze besluiten geen regels vooraf te formuleren, maar keuzes situationeel te bepalen. Principegebaseerde en regelgebaseerde architecturen hebben ieder hun voor- en nadelen en passen goed in verschillende contexten.

**Tabel 1.**  
Parallele ketens  
van principes

Business principe	1. Bedrijfsfuncties kunnen autonoom worden uitgevoerd	
Afgeleid IT-principe	1.1. Applicaties communiceren via services	
Afgeleide IT-regels	1.1.1. (A) Endpoint naming convention is gebaseerd op business functie	1.1.1. (B) Endpoint naming convention is gebaseerd op object en actie
	1.1.2. (A) Authenticatie is gebaseerd op mutual SSL	1.1.2. (B) Authenticatie is gebaseerd op user profile

Het IT-principe 1.1. 'Applicaties communiceren via services' wordt op twee verschillende manieren uitgewerkt in regels. Varianten A en B zijn beide detailleringen van 1.1., maar maken daarin verschillende keuzes.

## PRINCIPEGEBASEERD

Een principegebaseerde architectuur wil zeggen dat de architecten zich beperken tot het aanleveren van een aantal principes die duidelijk aangeven welke resultaten bereikt moeten worden, zonder het hoe daarvan in detail te geven. Dat hoe wordt overgelaten aan de eigenaren, ontwerpers en beheerders van de systemen. Dit legt verantwoordelijkheid over het hoe bij de uitvoerders en geeft de organisatie daarmee de mogelijkheid om snel in te spelen op externe ontwikkelingen. Ook borgt het dat de organisatie leert

## AUTEURS



**MARLIES VAN STEENBERGEN** (marlies.van.steenbergen@sogeti.com) is principal consultant enterprise architectuur bij Sogeti Nederland en is als onderzoeker verbonden aan de Hogeschool Utrecht.



**JAN BOUMAN** (jan.bouman@sogeti.com) is lead business analyst bij Sogeti Nederland en is adviseur-auteur over vraagstukken op het gebied van IT-management, -regie en -finance en over IT-aanbestedingen.



**TON EUSTERBROCK** (ton.eusterbrock@sogeti.com) is enterprise architect bij Sogeti Nederland.

om te gaan met onverwachte gebeurtenissen. Immers, men is gewent om binnen de kaders van de principes zelf na te denken over de precieze invulling. Met name in onvoorspelbare omgevingen is deze zelfstandigheid wenselijk.

**REGELGEBASEERD**

In een regelgebaseerde architectuur gaan de architecten verder in het uitwerken van de principes en schrijven ze ook het hoe voor. Dit geeft meer centrale controle, wat fijn is in situaties waar ruimte voor eigen interpretatie ongewenst is. In voorspelbare omgevingen is een regelgebaseerde architectuur veel efficiënter. Het risico van een regelgebaseerde architectuur is wel dat regels, door hun nadruk op het hoe, sneller achterhaald raken. Het niet tijdig herijken van regels kan leiden tot zogenaamde fossilisatie van kennis, waarbij een organisatie blijft handelen op basis van verouderde uitgangspunten. In een organisatie waar principegebaseerde architectuurvoorschriften de norm zijn, kunnen bewuste regels ervoor zorgen dat tunnelvisies of belemmerende conservatieve ideeën doorbroken worden.

**VERSCHILLENDE ARCHITECTUURREGIMES NAAST ELKAAR**

De diversiteit binnen organisaties neemt toe. Waar voorheen in de meeste organisaties het interne productieproces zaken als tempo, kwaliteitsregime, oplevering en besturingsfilosofie bepaalden, is dat tegenwoordig niet vol te houden. Door de veranderingen in de markt richten organisaties zich veel meer op de buitenwereld. Dit betreft zowel het samenwerken met partners als het omgaan met klanten. De diversiteit

“Het risico van een regelgebaseerde architectuur is dat regels sneller achterhaald raken.”

van de wereld dringt daarmee door in de organisatie.

Moderne organisaties hebben interne heterogeniteit nodig om succesvol te zijn. Deze heterogeniteit is nodig in processen, organisatiestructuur, informatie- en datamanagement, IT-systemen en infrastructuur. Het onderwerpen van de organisatie aan één uniform IT-architectuurregime doet geen recht aan deze noodzakelijke heterogeniteit. Ook in IT-architectuur is differentiatie geboden. Differentiatie in inhoud, werkwijze, vorm en besturing.

Gartner spreekt van bimodal IT en pace layering, waarbij onderscheid gemaakt wordt in ‘systems of record’, ‘systems of differentiation’ en ‘systems of innovation’. Dit onderscheid wordt gedreven door verschillen in snelheid van ontwikkeling. Echter, er zijn meer factoren die belangrijk zijn, zoals eisen aan betrouwbaarheid en beschikbaarheid, het bijpassende besturingsmodel of het type data waar het systeem mee werkt (zie tabel 2). Deze factoren worden gedreven door het businessdoel van het systeem en de rol die het systeem vervult in de omgeving. Zo hebben IT-systemen die tot doel hebben om diensten op innovatieve manieren toegankelijk te maken voor

**Tabel 2.**  
Kenmerken om IT-systemen te beschrijven.

KENMERK	BESCHRIJVING	IMPACT OP ARCHITECTUURREGIME
Variabiliteit	De levensduur en stabiliteit van de systemen en de frequentie van verandering	Hoe veranderlijker hoe minder geschikt voor centraal gedefinieerde alomvattende detailkaders
Veranderaanpak	De methoden, tools, katalysatoren en 'sourcing mix' waarmee nieuwe oplossingen worden gecreëerd, inclusief mogelijke cocreatie met klanten of partners	Hoe minder voorspelbaar, hoe meer behoefte aan een minimale benadering bij het stellen van architectuurkaders: just enough, just in time
Besturing	Het type controle dat wordt toegepast, inclusief eigenaarschap van ontwerpkeuzes en stakeholder betrokkenheid	Waar centrale sturing gewenst is, is meer ruimte voor gedetailleerde kaders
Betrouwbaarheid	De kwaliteitseisen aan de systemen, inclusief beveiligings- en continuïteitseisen	Verschillen in eisen vergen verschillende architectuurprincipes of regels
Bereik	De positie van het systeem in het gehele ecosysteem, inclusief koppelingen naar buiten	De positie in het ecosysteem bepaalt het belang van centraal geformuleerde architectuur.
Data	De hoeveelheid, vluchtigheid, structuur en actualiteit van de data	Diversiteit in data vraagt om verschillende architectuurprincipes of regels.

“Eén uniforme IT-architectuur doet geen recht aan de interne heterogeniteit.”

klanten een andere configuratie van factoren dan IT-systemen waarin langlopende contracten worden beheerd. De factoren samen bepalen voor een specifiek IT-systeem welk architectuurregime passend is. Welke varianten in de praktijk onderscheiden worden kan van organisatie tot organisatie verschillen. Met behulp van de kenmerken in tabel 2 kunnen IT-systemen getypeerd worden en architectuurregimes aan de typeringen gekoppeld worden.

Door bewust gebruik te maken van het onderscheid tussen principe- en regelgebaseerde architectuur kunnen architecten heel concreet invulling geven aan de verschillen in architectuurregimes: een multidynamische architectuur.

### SWEET SPOTS

Er is geen algemene uitspraak te doen of principegebaseerd beter is dan regelgebaseerd of andersom. Sterker nog, de heterogeniteit binnen een organisatie zal voor sommige delen van de organisatie om een principegebaseerde aanpak vragen en voor andere om een regelgebaseerde aanpak. Bij elk systeemtype hoort een eigen 'sweet spot'. Zo kan de architect onderscheid maken, niet alleen in inhoud, maar ook in besturing. Architecten kunnen via een aantal stappen deze sweet spots vinden en de multidynamische architectuurregimes daarop afstemmen.

#### Stap 1: Systeemtypen definiëren

Het definiëren van de te onderscheiden systeemtypen moet per organisatie uitgevoerd worden. Het is een kwestie van een aantal representatieve systemen beschrijven aan de hand van de zes kenmerken van tabel 2. Daaruit ontstaat een aantal configuraties, oftewel systeemtypen.

**Stap 2: Sweet spot per systeemtype vaststellen**

Op basis van de configuratie kan per systeemtype de sweet spot bepaald worden. Zo zal een systeemtype dat grote variabiliteit kent en waar de veranderaanpak voor een belangrijk deel leunt op cocreatie met klanten, een sweet spot hebben die ligt aan de principe kant van het continuüm. Immers, een dergelijk systeemtype vergt dat ontwerpers en ontwikkelaars kunnen experimenteren, dat nieuwe ontwikkelingen op de voet gevolgd kunnen worden en dat nauw kan worden aangesloten bij de klantbehoefte. Daar passen centraal geformuleerde regels niet bij en kan het best volstaan worden met enkele essentiële principes.

**Stap 3: Bestaande principes en regels vertalen naar de diverse architectuurregimes**

Doorgaans zal een organisatie al een verzameling principes en regels hebben. In dat geval is het nodig om deze verzameling te herijken en om te zetten naar sets per systeemtype. Een deel van de bestaande verzameling zal voor alle systeemtypen van de organisatie gelden. Maar van sommige regels kan besloten worden dat ze niet langer onder verantwoordelijkheid van de centrale architectuurfunctie vallen. Deze worden overgedragen aan het organisatieonderdeel waar ze betrekking op hebben. Het is vervolgens aan het organisatieonderdeel om ze ongewijzigd in de eigen werkwijze op te nemen als regels, ze aan te passen of om er afstand van te doen.

**Stap 4: Besturing inrichten**

Als de sweet spots zijn bepaald en de bestaande principes en regels aan de juiste verantwoordelijken zijn toege-

“Regels kunnen ervoor zorgen dat conservatieve ideeën doorbroken worden.”

kend, is het nog wel zaak om de besturing van het geheel expliciet in te richten.

**CONCLUSIE**

Het slim toepassen van principe- en regelgebaseerde architectuurvoor-schriften in combinatie met elkaar is een belangrijk kenmerk van multidy-namische architectuur. Verschillende IT-systemen hebben hun eigen sweet spot. Door dit expliciet te maken, blijft de besturing hanteerbaar, zonder geforceerde eenvormigheid, en zonder alles zomaar los te laten.

Als deze stap is gezet, kan met de opgedane ervaring de volgende stap gezet worden richting business-architectuur. Want dit is nog maar het begin. Het gaat namelijk niet om IT-systemen in isolatie, maar om systemen in termen van mensen, processen en technologie gezamenlijk die een bepaalde rol hebben binnen een organisatie. Dan maken we de stap van systeemtypen naar systemen in brede zin en daarmee van IT-architectuur naar enterprise-architectuur. 